What is claimed is:

1. A method of encoding three-dimensional object data, which is comprised of point texture data, voxel data, or octree structure data, the method comprising:

generating three-dimensional object data having a tree structure in which nodes are attached labels indicating their types;

encoding nodes of the three-dimensional object data; and

generating the three-dimensional object data whose nodes are encoded into a bitstream.

2. The method of claim 1, wherein in generating the three-dimensional object data, voxels are differentiated from one another by converting the three-dimensional object data into voxel data using three-dimensional bounding volume and differently labeling the voxels depending on whether or not they are located at places where objects exist.

3. The method of claim 2, wherein in the tree structure representing the three-dimensional object data, a node having sub-nodes is labeled ‘S’, a node whose voxels do not contain objects is labeled ‘W’, a node whose voxels all contain objects is labeled ‘B’, and a node whose voxels are encoded using a prediction-by-partial-matching (PPM) algorithm is labeled ‘P’.

4. The method of claim 1, wherein encoding the nodes of the three-dimensional object data comprises:

encoding node information which indicates whether or not a current node is an ‘S’ node or a ‘P’ node; and

encoding detailed information bit (DIB) data of an ‘S’ node if the node information indicates that the current node is an ‘S’ node and encoding DIB data of a ‘P’ node if the node information indicates that the current node is a ‘P’ node.

5. The method of claim 1, wherein in encoding the nodes of the three-dimensional object data, only some of the nodes of the three-dimensional object data, ranging from a root node to a predetermined lower node, are encoded.

6. The method of claim 1, wherein in encoding the nodes of the three-dimensional object data, all the nodes of the three-dimensional object data are encoded.

7. A method of encoding three-dimensional object data, which is comprised of point texture data, voxel data, or octree structure data, the method comprising:

(a) generating three-dimensional object data having a tree structure in which nodes are attached labels indicating their types;

(b) merging the nodes of the three-dimensional object data by referring to their labels;

(c) encoding merged nodes;

(d) generating the three-dimensional object data whose merged nodes are encoded into a bitstream; and

(e) repeatedly carrying out steps (a) through (d) until an uppermost node of the tree structure representing the three-dimensional object data is encoded.

8. The method of claim 7, wherein in step (a), a node having sub-nodes is labeled 'S', a node whose voxels do not contain objects is labeled 'W', a node whose voxels all contain objects is labeled 'B', and a node whose voxels are encoded using a PPM algorithm is labeled 'P'.

9. The method of claim 8, wherein step (b) comprises:

selecting 'P' nodes and 'S' nodes whose sub-nodes are labeled 'W' and 'B' as candidate nodes to be merged;

selecting, from among the candidate nodes as an optimal node, a node which can minimize a ratio of a difference ΔD between the number of distorted bits before merging the candidate nodes and the number of distorted bits after merging the candidate nodes with respect to a difference ΔR between the number of bits before merging the candidate bits and the number of bits after merging the candidate bits;

labeling the selected node 'B'; and

updating all the candidate nodes except the node selected as an optimal node.

10. The method of claim 9, wherein D is calculated in the following equation using a Hamming distance between an original model V and its approximation $\widehat{V}$ as distortion measurement:

$$D=\sum_{x=1}^{X}\sum_{y=1}^{Y}\sum_{z=1}^{Z}\left|V(x,y,z)-\widehat{V}(x,y,z)\right|$$

where X×Y×Z represents the resolution of the original model.

11. The method of claim 8, wherein step (c) comprises:

encoding a continue flag which is information indicating whether or not the candidate nodes exist in a queue;

encoding node position information which indicates position of each of the candidate nodes in the queue;

encoding node type information which indicates whether or not a current node is an 'S' node or a 'P' node; and

encoding DIB data of an 'S' node if the node information indicates that the current node is an 'S' node and encoding DIB data of a 'P' node if the node information indicates that the current node is a 'P' node.

12. The method of claim 11, wherein encoding the DIB data of the 'S' node comprises:

encoding an average of color information; and

encoding labels of eight sub-nodes of the current node.

13. The method of claim 11, wherein encoding the DIB data of the 'P' node comprises:

encoding depth information; and

encoding the color information.

14. The method of claim 13, wherein in encoding the depth information, all nodes below a predetermined node in the tree structure representing the three-dimensional object data are PPM-encoded according to a raster scanning order by using a predetermined number of contexts.

15. The method of claim 13, wherein in encoding the color information, red (R), green (G), and blue (B) values of 'B' voxels of the current node are encoded by carrying out differential pulse code modulation (DPCM) and adaptive arithmetic coding (AAC).

16. The method of claim 7, wherein in step (c), only some of the merged nodes, ranging from a first one to a predetermined numbered one, are encoded.

17. The method of claim 7, wherein in step (c), all the merged nodes are encoded.

18. An apparatus for encoding three-dimensional object data, which is comprised of point texture data, voxel data, or octree structure data, the apparatus comprising:

a tree structure generator which generates three-dimensional object data having a tree structure in which nodes are attached labels indicating their types;

a merging order selector which merges the nodes of the three-dimensional object data by referring to their labels;

a node encoder which encodes merged nodes; and

a bitstream generator which generates the three-dimensional object data whose merged nodes are encoded into a bitstream.

19. The apparatus of claim 18, wherein the merging order selector comprises:

a candidate node selector which selects 'P' nodes and 'S' nodes whose sub-nodes are labeled 'W' and 'B' as candidate nodes to be merged;

an optimal node selector which selects, from among the candidate nodes as an optimal node, a node which can minimize a ratio of a difference D between the number of distorted bits before merging the candidate nodes and the number of distorted bits after merging the candidate nodes with respect to a difference R between the number of bits before merging the candidate bits and the number of bits after merging the candidate bits and labels the selected node 'B'; and

a candidate node updater which updates all the candidate nodes except the node selected as an optimal node.

20. The apparatus of claim 18, wherein the node encoder comprises:

a continue flag encoder which encodes a continue flag which is information indicating whether or not a current node is the end of a compressed bitstream;

a node position encoder which encodes node position information indicating position of each of the candidate nodes in the queue;

a node 'S'-or-'P' (SOP) selector which encodes node type information indicating whether or not the current node is an 'S' node or a 'P' node;

an S node encoder which encodes DIB data of an 'S' node if the node information indicates that the current node is an 'S' node; and

a P node encoder which encodes DIB data of a 'P' node if the node information indicates that the current node is a 'P' node.

21. The apparatus of claim 18, wherein the node encoder encodes only some of the merged nodes, ranging from a first one to a predetermined numbered one, are encoded.

22. The apparatus of claim 18, wherein the node encoder encodes all the merged nodes.

23. A method of decoding three-dimensional object data, comprising:

reading continue flag information from a bitstream of encoded three-dimensional object data and decoding the continue flag information;

decoding note type information of the bitstream;

decoding an 'S' node if the note type information indicates that a current node is an 'S' node and decoding a PPM node if the note type information indicates that the current node is a PPM node; and

restoring the three-dimensional object data whose nodes are encoded to a tree structure.

24. A method of decoding three-dimensional object data, comprising:

decoding nodes of a bitstream of encoded three-dimensional object data; and

restoring the three-dimensional object data whose nodes are encoded to a tree structure.

25. The method of claim 24, wherein decoding the nodes of the bitstream of the encoded three-dimensional object data comprises:

reading continue flag information from a bitstream of encoded three-dimensional object data and decoding the continue flag information;

reading node position information indicating which candidate node in a queue is a current node and decoding the node position information;

decoding note type information of the bitstream;

decoding an 'S' node if the note type information indicates that a current node is an 'S' node; and

decoding a PPM node if the note type information indicates that the current node is a PPM node.

26. The method of claim 25, wherein in decoding the 'S' node, an average color of eight sub-nodes of the current node is decoded as DIB data, and the eight sub-nodes are sequentially decoded into black nodes ('B' nodes) or white nodes ('W' nodes).

27. The method of claim 26, wherein in decoding the PPM node, the current node is PPM-decoded using DIB data bits (DIB) data, and R, G, and B values of 'B' voxels of the current node are decoded by carrying out inverse AAC and inverse DPCM.

28. An apparatus for decoding a bitstream of encoded three-dimensional object data, the apparatus comprising:

a bitstream reader which receives a bitstream of encoded three-dimensional object data;

a node decoder which decodes the bitstream; and

a tree structure restorer which restores decoded nodes to a tree structure.

29. The apparatus for claim 28, wherein the node decoder comprises:

a continue flag decoder which decodes a continue flag indicating whether or not a current node is the end of the bitstream;

a node position information decoder which reads node position information indicating which candidate node in a queue is a current node and decoding the node position information;

a node type selector which decodes note type information of the bitstream;

an S node decoder which decodes an average color of eight sub-nodes of the current node as DIB data and then sequentially decodes the eight sub-nodes into 'B' nodes or 'W' nodes; and

a P node decoder which PPM-decodes DIB data of the current node and then decodes R, G, and B values of 'B' voxels of the current node by carrying out inverse AAC and inverse DPCM decoding an 'S' node, if the note type information indicates that a current node is a PPM node.

30. A computer-readable recording medium on which a program enabling the method of claim 1 is recorded.

31. A computer-readable recording medium on which a program enabling the method of claim 7 is recorded.

32. A computer-readable recording medium on which a program enabling the method of claim 23 is recorded.

33. A computer-readable recording medium on which a program enabling the method of claim 24 is recorded.